(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 903 895 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
24.03.1999 Bulletin 1999/12

(51) Int Cl.[6]: **H04L 12/56**

(21) Application number: **98306787.7**

(22) Date of filing: **25.08.1998**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **02.09.1997 US 56028 P**

(71) Applicant: **LUCENT TECHNOLOGIES INC.**
Murray Hill, New Jersey 07974-0636 (US)

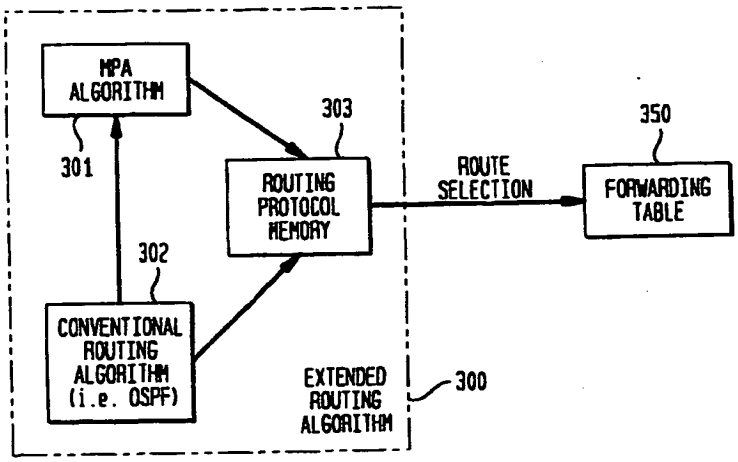
(72) Inventors:
- **Narvaez-Guarnieri, Paolo**
  Cambridge, MA 02139 (US)
- **Tzeng, Hong-Yi**
  Trenton Falls, NJ 07712 (US)
- **Siu, Kai-Yeung**
  Charlestown, MA 02129 (US)

(74) Representative:
Watts, Christopher Malcolm Kelway, Dr. et al
Lucent Technologies (UK) Ltd,
5 Mornington Road
Woodford Green Essex, IG8 0TU (GB)

(54) Multiple path routing

(57) A novel data structure in a router helps to compute viable next hops for forwarding a data packet from a router to its destination along multiple alternate loop-free paths, which are not necessarily of shortest distance. Each viable next hop may also be specified with a degree of optimality, which enables a route to perform QoS routing and fault-tolerant routing efficiently. The data structure can be implemented as an add-on software to existing routing protocols and may be implemented in existing networks which use shortest path protocols, even where less than all of the routers use the data structure and multiple path scheme described herein.

FIG. 3



EP 0 903 895 A2

Printed by Jouve, 75001 PARIS (FR)

Description

RELATED APPLICATIONS

[0001] This application claims the benefit of provisional U.S. Patent Application Serial No. 60/056,027 filed September 2, 1997.

FIELD OF THE INVENTION

[0002] The present invention relates generally to routing data packets to a destination node within a packet switching network. In particular the invention relate to a multiple path routing scheme using a novel data structure.

BACKGROUND OF THE INVENTION

[0003] In packet switching networks such as the Internet, a packet from a source traverses intermediate network routers before reaching its destination. Before a packet arrives at a router, a routing protocol is used to determine the existence and status of all routers and links therebetween on the network. This information is used by the router to determine an appropriate next hop for an arriving packet with a given destination. The next hop includes the next router along the path to the destination and the outgoing port of the present router linking the next router. A widely used link-state routing protocol, is Open Shortest Path First ("OSPF")

[0004] Given the current link states in the network, (e.g., reduced bandwidth, increased bandwidth, disconnected, reconnected) each router S keeps track of a shortest path tree "SPT" from S to every other router in the network. Since the router retains only one shortest path from S for each destination router, the first hop from S along this path will be used as the next hop to forward any packets with the same given destination.

[0005] Although a router using OSPF has global information about the regional network topology, it determines only the local route of a packet, i.e., the next hop. When each router computes such next-hop information based on an SPT, each packet is guaranteed to be forwarded along a loop-free path to its destination. A loop-free path is a path that doesn't include the same router more than once. Multiple SPTs rooted at router S can co-exist in a network. In such a case there are multiple choices of next hops from S to forward a packet, each of which will lead to a different path of the same shortest distance. The packet can be forwarded to any of these next hops.

[0006] Routing protocols allowing for only a single shortest path suffer from several inefficiencies. For example, if immediately before dispatching a packet along its next hop, a router learns that a link along the path is down, the router must first recompute a new single shortest path and only then send the packet along the new next hop.

[0007] The single shortest path method also ignores network load considerations. Thus, even if a path has a lot of traffic the packet will be forwarded along that path if it is the shortest to its destination. Furthermore, it is known that finding an optimal path that satisfies certain QoS constraints is in general computationally difficult (NP-complete). Since some QoS parameters that need to be optimized may depend on the network traffic load and thus can change dynamically over a short period of time, finding an optimal path can become infeasible even for a moderate-size network. Even for QoS parameters that do not change frequently, determining the optimal path from a router may require global knowledge of these parameters in the network, and a local change in these parameters would therefore need to be disseminated to every node in the network. Hence in most cases only suboptimal paths can be computed.

SUMMARY OF THE INVENTION

[0008] The present invention uses a multiple path algorithm (referred to herein as "MPA") to enable a router to forward packets to multiple viable next hops that are not strictly constrained to be a part of a shortest path from the router to the packet's destination. The method of the present invention still guarantees, however, that all packets will be routed to their destination on loop-free paths. In general each path traversing a different next hop is guaranteed to be loop free if it satisfies the constraint that the shortest distance from the implementing router to the destination decreases at each next hop.

[0009] In a preferred embodiment of the invention, the MPA uses a novel data structure which stores information relating to each router (node) in the network that is a potential destination node. In particular, the data structure maintains at least the following attributes: the shortest distance from the implementing router to the destination node; the cost of each link from the implementing router to each potential next hop; and for each potential next hop, the shortest distance from the implementing router to the destination node along a path traversing that particular next hop.

[0010] Using this data structure, multiple viable next hops are computed by selecting next hops from the potential next hops whose shortest distance attribute traversing that next hop, minus the cost of the link from the implementing router to the next hop, is less than the shortest path from the implementing router to the destination node.

[0011] The method and data structure of the present invention can be efficiently implemented as an add-on component to existing routing protocols such as OSPF. Each MPA implementing router uses the topology information exchanged by the router protocols to compute multiple paths between a given source and a given destination. Moreover, conventional routers based on shortest path routing can co-exist in the same network with MPA implementing routers to guarantee loop-free rout-

ing for each packet. In general, an MPA implementing router can interoperate with any router whose routing protocol uses the loop-free policy.

[0012] By maintaining multiple viable next hops in a router for each destination, recovery from link failure is much faster than heretofore possible, since the recovery mechanism using alternate paths can be implemented locally by the router, and there is almost no delay incurred by the link failure. On the other hand, without MPA, after detecting a link failure, the router would need to broadcast throughout the network the link failure, then recompute the shortest path tree to determine a new alternative path.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013]

FIG. 1 is a graphical depiction of the data structure used in accordance with the present invention.

FIG. 2 is a graphical depiction of the data structure used in a conventional shortest path tree algorithm.

FIG. 3 depicts a router architecture employing the MPA algorithm as used in accordance with the present invention.

FIG. 4 is a representation of a network of routers including a source router S and a destination router D.

DETAILED DESCRIPTION OF THE INVENTION

[0014] The packet routing method of the present invention involves making available at each router, alternative multiple next hops to forward a received packet. The only constraint in computing the multiple next hops is that the paths to the packet's destination resulting from these next hops do not return the packet to any router it previously hit. In other words, the paths of these alternative next hops should be loop free. As discussed above, the OSPF protocol well known in the art, guarantees loop-free routing to each destination by constraining each router to choose the next hop based on an SPT. However, a loop free path can be guaranteed even if it does not represent the shortest distance to a destination. For example, any path to a destination that traverses a particular next hop from a router S is loop free if the distance to the destination continues to decrease at each next hop.

[0015] In one preferred embodiment of the present invention viable next hops from a router S, i.e., next hops which lie on a loop free path to a destination, can be guaranteed where the shortest distance from S to the destination traversing a given next hop, minus the cost of the link (i.e., its distance) from S to that next hop is less than the absolute shortest distance from S to the destination. For example, if router R is the next hop from router S along a shortest path from S to destination router D, then the distance between R and D, i.e., d(R, D), must be less than the distance from S to D, i.e., d(S, D). Extrapolating R to any next router R_i, which can even be a router not on the absolute shortest path between S and D, a loop free path from S to D is guaranteed with a next hop to R_i if $d(R_i, D) < d(S, D)$.

[0016] Referring to FIG. 4 an example of a network of routers is shown, including source router S, destination router D and a plurality of routers R. Each link between the routers is numbered and it is assumed that each link has the same cost, or distance. In this example, it is clear that there are three shortest paths: (i) 1-2-3; (ii) 7-9-10; and (iii) 4-14-3.

[0017] However, in addition to these shortest paths, there are at least twenty four loop free paths. In particular, 1-5-6-14-3; 1-5-6-15-16-3; 1-5-6-15-12-13; 1-5-6-8-9-10; 1-5-6-8-9-11-16-3; 1-5-6-8-9-11-12-13; 4-5-2-3; 4-6-15-16-3; 4-6-15-12-13; 4-6-15-11-10; 4-6-8-9-10; 4-6-8-9-11-16-3; 4-6-8-9-11-12-13; 7-9-11-16-3; 7-9-11-12-13; 7-8-15-16-3; 7-8-15-11-10; 7-8-15-12-13; 7-8-6-14-3; 7-8-6-14-16-12-13; 7-8-6-14-16-11-10; 7-8-6-5-2-3; 7-8-6-5-2-16-11-10; 7-8-6-5-2-16-12-13. A non-viable loop path, however, exists for example on path 7-8-6-5-2-16-15. . . since link 15 will return the packet to a router previously visited between links 8 and 6.

[0018] In the preferred embodiment of the invention, the MPA of the present invention makes use of the data structure of FIG. 1 to compute alternate viable next hops from a router whose shortest distances to the destination are less than the shortest distance from the router to the destination.

[0019] During execution, a conventional SPT algorithm usually keeps track of certain state information about each node in the network. See E. Dijkstra, "A Note to Two Problems in Connection with Graphs" Numerical Mathematics, vol. 1 p. 115-126 (1959); R. Bellman, "On a Routing Problem" Quarterly of Applied Mathematics, vol. 16, p. 87-90 (1958), hereby incorporated by reference as if fully set forth herein. Typically, such information consists of a distance attribute 210, the next hop attribute from the router 220, and perhaps a parent attribute for each node 230, as shown in FIG. 2. At each hop, during the execution of the SPT algorithm in router S, the distance attribute 210 of destination node X represents the distance of the shortest path found so far from the router S to node X, while the next hop attribute and the parent attribute respectively identify the first hop from router S and the last hop to router S.

[0020] The data structure of FIG. 1, however, contains more state information about each node in the network, without incurring significant computational complexity. More specifically, a given router S will maintain for each destination node X the shortest distance d(S, X) from the router S to X. This information can be computed with existing SPT algorithms such as those taught in Dijkstra and Bellman, supra. In addition, the distance, or cost w (S,p) for each outgoing link connecting the router S to a potential next hop p is also maintained. Furthermore, the

data structure keeps track of the length of the shortest path that uses p as the next hop from S, which is denoted by $d_p(S, X)$. When the MPA execution terminates, $d_p(S, X)$ should equal the sum of w(S,P) and the shortest distance found, d(p, X) from p to X. By maintaining this data structure, the MPA identifies viable next hops for destination X using the principles discussed above: namely a viable path using next hop p exists if $d_p(S, X) - w(S, p) < d(S, X)$.

[0021] While all next hops satisfying the above inequality are viable and can be revealed by the MPA, it is possible for additional viable next hops to exist which will not be revealed. By searching through paths, the data structure at every node can be updated, increasing the number of viable next hops. As long as the shortest distance to every node is computed correctly, the resulting viable next hops will not lead to a loop.

[0022] With the alternative viable paths for a given destination, alternate paths can be constructed by comparing the MPA data structures associated with different destinations. Most SPT algorithms known in the art constantly compare the distance attributes of two neighboring nodes. The distance attribute of a node eventually converges to the shortest distance from the source to that node. The distance attribute of node X from source S computed by the SPT algorithm is contained in the corresponding field 210 of the shortest distance from S to X and is updated by the SPT algorithm, independently of other operations of the MPA.

[0023] Referring to FIG. 3, the extended routing algorithm 300 in accordance with the present invention comprises these modules: MPA Algorithm 301, Conventional Routing Algorithm 302 and Routing Protocol Memory 303. The MPA and Conventional Algorithms 301 and 302, respectively, while possible to be implemented with hardware, typically comprise software executed on a standard microprocessor having the computing power of at least a Power PC or Pentium grade chip. Routing Protocol Memory 303 typically comprises at least 4 megabytes ("MB") of RAM, with 3MB typically allocated to the Conventional Algorithm 302 and 1MB allocated to the MPA Algorithm. Furthermore, it is possible to provide physically separate RAM chips for use by each of the algorithms 301 and 302. Forwarding Table 350 also typically comprises at least 4MB of RAM.

[0024] Every time the SPT algorithm, represented by block 302, makes a comparison between nodes a and b, the distance attribute from source S to b, $d_{opt}(b)$, is equal to the smaller values between its old value and the sum of $d_{opt}(a) + w(a, b)$.

[0025] Furthermore, with each comparison by the SPT algorithm 302 between nodes a and b, MPA block 301 is triggered and starts its own comparisons. For every next hop p from S, the distance from S to b is set to the minimum of its old value and $d_p(S, a) + w(a, b)$. With the termination of the SPT algorithm, the MPA also terminates and the data structure is updated in routing protocol memory 303.

[0026] With the availability of viable next hops the router can select the actual next hop for forwarding a packet based on a variety of criteria such as round robin, network load, or shortest distance. Those skilled in the art will be able to devise various selection criteria.

[0027] Indeed, one such criteria can be used to determine satisfactory approximate solutions to QoS routing by decoupling the underlying optimization problem discussed above, into local and global computations. First, multiple viable paths that satisfy the global or less frequently changed QoS constraints are computed. Then among these viable paths, one that is optimized with respect to the local or more frequently changed QoS parameters can be determined.

[0028] With respect to load balancing it should be noted that implementation of such a scheme may require a means to preserve the FIFO arrival of packets where FIFO is assumed by the upper layer protocol at the destination router. It may also be necessary to provide additional buffer means and processing capabilities at the destination router, especially where FIFO is not strictly adhered to.

[0029] In addition each router can compute and store in a forwarding table such as 350 in FIG. 3, each viable next hop for a given destination or maintain only a single viable next hop in forwarding table 350 while maintaining additional next hop information in the routing protocol memory 303.

[0030] By maintaining multiple viable next hops in a router for each destination, recovery from link failure is much faster than heretofore possible, since the recovery mechanism using alternate paths can be implemented locally by the router, and there is almost no delay incurred by the link failure. On the other hand, without MPA, after detecting a link failure, the router would need to broadcast throughout the network the link failure, then recompute the shortest path tree to determine a new alternative path.

[0031] A more exhaustive search of alternate next hops can be obtained by running a set of independent SPT algorithms in parallel. For every next hop p_i any SPT algorithm including a dynamic algorithm, i.e., an algorithm which can compute a new SPT by readjusting the old SPT, is executed using only the distance attribute $d_p(S,X)$ in the data structure associated with the destination X. (While a static algorithm recomputes an entire shortest path, a dynamic algorithm recognizes that the majority of the path segments will remain the same and therefore only certain segments are recomputed). After the SPT algorithm is executed, the data structure is updated and all of the next hops p_i that satisfy the above inequality are obtained.

Claims

1. A method for routing a data packet from a router to a destination node in a packet switching network,

said data packet originally from a source node in said packet switching network, said method comprising the steps of:

determining one or more viable next hops from said router, each of said viable next hops lying on a path from said source node to said destination node, said path having a distance which decreases at each next hop along said path until said destination node;

storing in a first memory means at said router said one or more viable next hops; and

selecting a first one of said one or more viable next hops to forward said data packet.

2. A method according to Claim 1 wherein each of said viable next hops are in a path from said destination whose shortest distance is less than the absolute shortest distance from said source node to said destination node.

3. A method according to Claim 1 further comprising the step of selecting a second one of said one or more viable next hops if it is determined that a link failure exists along said path of said first viable next hops and a second viable next hop is stored in said first memory means.

4. A method according to Claim 1 wherein each of said viable next hops leads to a loop free path from said source node to said destination node, but is not constrained to lead to a shortest path between said source node and said destination node.

5. A method according to Claim 1 wherein said selecting steps are constrained to select a viable next hop leading to the shortest path as compared with all of said one or more viable next hops stored in said first memory means.

6. A method for determining multiple loop free paths from a source node to a destination node in a packet switching network, said packet switching network comprising a plurality of routers, said method comprising the step of selecting at each of said routers, one or more viable next hops for forwarding a data packet originating from a source node, to a destination mode, each of said viable next hops constrained by the inequality that the distance from said next hop to said destination node is less than the shortest distance from said source node to said destination node.

7. A router for use in a packet switching network for routing a data packet originating from a source node, to a destination node, comprising:

a data structure comprising,

the shortest distance from said router to a destination node;

the distance to each next hop from said router; and

for each of said next hops, the distance of the shortest path from said router to said destination node using said next hop;

computer implemented means for determining one or more viable next hops from said data structure, each of said viable next hops satisfying the inequality that the difference between the distance from said router to said destination node traversing one of said viable next hops, and the cost from said router to said next hop is less than the shortest distance from said router to said destination node.

8. A data structure for use in a router in a packet switching network, comprising:

the shortest distance from said router to a destination node;

the cost of each next hop from said router; and

for each of said next hops, the distance of the shortest path from said router to said destination node using said next hop.

FIG. 1

SHORTEST DISTANCE FROM ROUTER S TO NODE X	HOP A	w(S,A)	$d_A(S,X)$
	HOP B	w(S,B)	$d_B(S,X)$
	HOP C	w(S,C)	$d_C(S,X)$
	HOP D	w(S,D)	$d_D(S,X)$
	⋮	⋮	⋮

FIG. 2
(PRIOR ART)

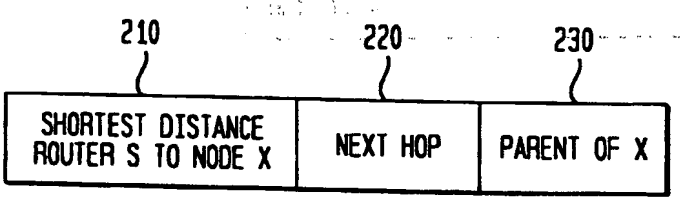


FIG. 3

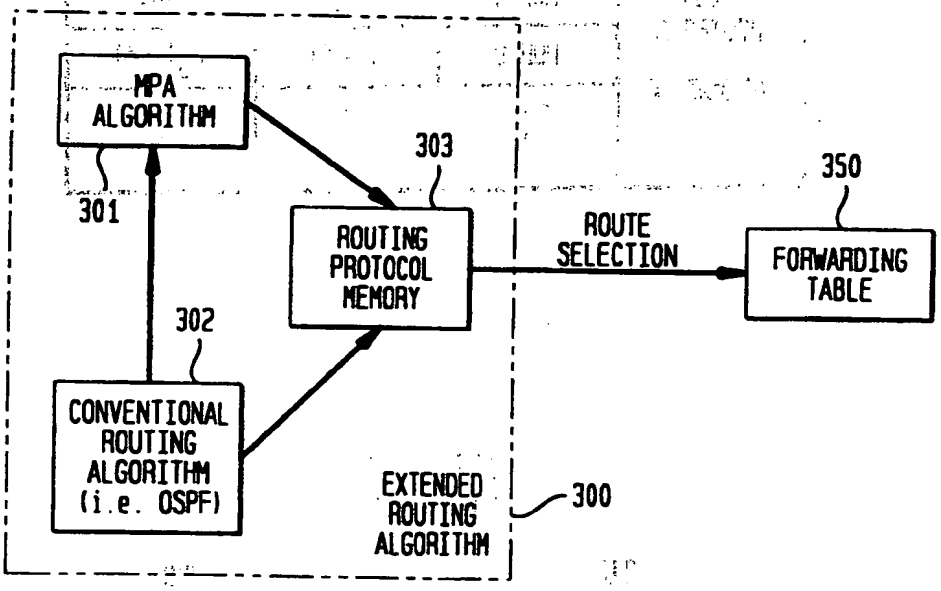

MPA ALGORITHM
301
303
350
ROUTING PROTOCOL MEMORY
ROUTE SELECTION
FORWARDING TABLE
302
CONVENTIONAL ROUTING ALGORITHM (i.e. OSPF)
EXTENDED ROUTING ALGORITHM
300

FIG. 4